

On state complexity of unions of binary factor-free languages

Szabolcs Iván^a

^aUniversity of Szeged, Hungary

Abstract

In [1], it has been conjectured that if K and L are factor-free regular languages over a binary alphabet having state complexity m and n , resp, then the state complexity of $K \cup L$ is at most $mn - (m + n) + 3 - \min\{m, n\}$. We disprove this conjecture by giving a lower bound of $mn - (m + n) - 2 - \lfloor \frac{\min\{m, n\} - 2}{2} \rfloor$, which exceeds the conjectured bound whenever $\min\{m, n\} \geq 10$.

1. Introduction

The state complexity of a regular language is the number of states of its minimal automaton. The state complexity of a binary operation \circ is a function on n, m : if K has state complexity n and L has state complexity m , then how large can the state complexity of $K \circ L$ be? In this note we give a lower bound for the state complexity of *union* when the alphabet is *binary* and the languages K, L are *factor-free*.

For a survey on state complexity see [6], for applications of (binary) factor-free languages in cryptography and coding theory see [5], for results on the state complexity of operations on subclasses of regular languages see [3, 2, 4, 1].

2. Notation

We assume the reader has basic knowledge on language and automata theory. An alphabet is a finite nonempty set Σ , a Σ -word is a finite sequence $w = a_1 \dots a_n$ of letters $a_i \in \Sigma$, with ε denoting the empty word when $n = 0$, a language (over Σ) is any set of Σ -words. The set Σ^* of all words forms a monoid with the operation being (con)catenation, or simply product of words given by $a_1 \dots a_n \cdot b_1 \dots b_k = a_1 \dots a_n b_1 \dots b_k$. A finite automaton is a system $M = (Q, \Sigma, \delta, q_0, F)$ with Q being the finite set of states, $q_0 \in Q$ the start state, $F \subseteq Q$ the set of final states, Σ is the finite nonempty input alphabet and $\delta : Q \times \Sigma \rightarrow Q$ the transition function which is extended to $Q \times \Sigma^* \rightarrow Q$ as $\delta(q, \varepsilon) = q$ and $\delta(q, wa) = \delta(\delta(q, w), a)$. If M is understood, we simply write qw for $\delta(q, w)$. The language recognized by M is $L(M) = \{w \in \Sigma^* : q_0 w \in F\}$.

States of the form $q_0 w$ are called *reachable* states of M . A trap is a non-final state $p \notin F$ such that $pa = p$ for each $a \in \Sigma$ (thus, $pw = p$ for each word w as well). Two states p, q are called *distinguishable* if there exists a word w such that exactly one of the states pw and qw belongs to F . It is known that M is minimal (that is, has the smallest possible number of states among the automata recognizing $L(M)$) iff each pair $p \neq q$ of its states is distinguishable and all its states are reachable. Such automata are also called *reduced*. A state q is empty if there is no word w with $qw \in F$. In a minimal automaton, there is at most one empty state which is then a trap. The state complexity of a regular (that is, recognizable by some finite automaton) language is the number of states of its minimal automaton.

A word u is a factor of a word v if $v = xuy$ for some words x and y . A language L is factor-free if $u, xuy \in L$ imply $x = y = \varepsilon$ for any $u, x, y \in \Sigma^*$.

3. Factor-free languages

In [1] the following lower bound was given:

Proposition 1. *For the binary alphabet $\Sigma = \{a, b\}$ there exist regular factor-free languages K_n and L_m for each $n, m > 6$ such that K_n has state complexity n , L_m has state complexity m and $K_n \cup L_m$ has state complexity $nm - (n + m) + 3 - \min\{n, m\}$.*

They conjectured this bound to be tight. In the rest of this paper we give a better lower bound, at least for the case when m, n are large enough. We note that due to [1], for *at least ternary* alphabets, $mn - (n + m)$ is a tight bound.

In this section we assume that m and n are large enough, say $\min\{m, n\} \geq 10$.

We define two reduced n -state automata A_n and B_n as follows. In both cases, the set of states is $[n] = \{1, \dots, n\}$, the start state is 1, the unique accepting state is $n - 1$, state n is a trap state and $p\sigma = n$ for $p \in \{n - 1, n\}$ and $\sigma \in \{a, b\}$. Moreover, $1a = 2$ and $1b = n$ in both cases.

For A_n , let $pa = p$ for each *odd* $3 \leq p < n - 1$, $pa = n$ for each *even* $2 \leq p < n - 1$, let $pb = p + 1$ for each $2 \leq p < n - 2$ and $pb = 2$ for $p = n - 2$.

For B_n , let $pa = p + 1$ for each $2 \leq p \leq n - 1$ and $pb = p$ for each $2 \leq p < n - 1$.

See Figure 3 for A_{10} (depicted vertically on the left) and B_7 (depicted horizontally on top). Missing edges all go to the trap state n .

It is clear that all states of A_n and B_n are reachable. A_n is also reduced: $ab^{n-4}a$ is accepted exactly from 1, and for each $2 \leq p \leq n - 2$, $b^{n-2-p}a$ is a word accepted exactly from p and not from any other state. The empty word is accepted only from $n - 1$, and n is a trap state, thus each pair of states is distinguishable.

Also, B_n is reduced: for each state $1 \leq p \leq n - 1$, a^{n-1-p} is a word which is accepted exactly from p and not from any other state, and $p = n$ is a trap state, thus again, each pair of states is distinguishable.

Let K_n stand for $L(A_n)$ and L_n stand for $L(B_n)$. We claim that K_n and L_n are factor-free. To see this, we start with a handy lemma (which is probably folklore but we were not able to find it in the literature¹).

Lemma 1. *A reduced automaton $A = (Q, \Sigma, \delta, q_0, F)$ with $|Q| > 1$ recognizes a factor-free language if and only if it satisfies all the following conditions:*

- i) q_0 has indegree 0;
- ii) $F = \{q_f\}$ is a singleton;
- iii) there is a trap state $\perp \neq q_f$ with $p\sigma = \perp$ for every $p \in \{q_f, \perp\}$, $\sigma \in \Sigma$;
- iv) for any word $u \in L(A)$ and state $p \in Q - \{q_0\}$ it holds that $pu = \perp$.

PROOF. Assume the conditions all hold for A and let $u, v, w \in \Sigma^*$ be words with $v \in L$ and $uvw \in L$. We show that $u = w = \varepsilon$ in this case, proving the factor-freeness of L . Suppose $u \neq \varepsilon$. Then by i), $q_0u \neq q_0$. By iv), $q_0uv = \perp$ then and thus by iii), $q_0uvw = \perp \notin F$ by ii) and that $q_f \neq \perp$. Hence $u = \varepsilon$. Supposing $w \neq \varepsilon$ we get that $q_0uvw = q_0vw = q_fw$ which is \perp by iii). Thus $w = \varepsilon$.

For the other direction, let L be a factor-free regular language, and let A be its minimal automaton. Suppose A does not satisfy i). Then since A is reduced, $q_0u = q_0$ for some nonempty word u . Since $|Q| > 1$ and A is reduced, F is nonempty, hence $q_0v \in F$ for some v . Thus, $uv \in L$ as well as its proper factor v , a contradiction. Thus A satisfies i).

Now suppose there are at least two final states p and q with $pu = q$ for some (nonempty) word u . Then since A is reduced, $p = q_0v$ for some word v , hence both vu and its proper factor v are in L , a contradiction. Hence, from a final state no other final state is reachable. This means that any state reachable from a final state by a nonempty word is an empty state. Since A is reduced, this yields that there is a trap state $\perp \notin F$ and $p\sigma = \perp$ for any final state $p \in F$ and letter $\sigma \in \Sigma$. Thus all the final states are indistinguishable, hence there is exactly one final state, thus A satisfies ii) and iii).

Also, if $pu = q \neq \perp$ for some $u \in L$ and $p \in Q - \{q_0\}$, then since A is reduced, $p = q_0v$ for some nonempty word v and q is nonempty, say $qw \in F$ for some w . Then both vuw and its proper factor u is in L , a contradiction. Hence A satisfies iv).

¹We would be grateful to the reviewers to suggest a source for this lemma.

Now it is easy to show that K_n and L_n are factor-free.

For K_n , we have that A_n clearly satisfies Conditions i), ii) (with $q_f = n - 1$) and iii) (with $\perp = n$) of Lemma 1. For iv), any word $u \in L_n$ has the form $u = awa$ with $|w|_b \equiv -1 \pmod{n-3}$. Since $ka = \perp$ for all even $2 \leq k < n - 2$, $ku = \perp$ for those states. Also, $kawa = \perp$ for $k \geq n - 2$ since any word of length at least two maps those states to \perp . Finally, if $3 \leq k < n - 2$ is odd, then kaw is either \perp or $k - 1$ which is an even number between 2 and $n - 3$ (inclusive). For such states $k - 1$ we have $(k - 1)a = \perp$, thus in this case $ku = \perp$ also holds, thus Condition iv) is also verified.

For L_n , observe that any word belonging to L_n has the form aua with $|u|_a = n - 4$, hence in particular any word of L_n contains exactly $n - 2$ number of a 's and each proper factor has less number of a 's, showing factor-freeness of L_n .

Theorem 1. *The state complexity of $K_n \cup L_m$ is $mn - (m + n) - 2 - \lfloor \frac{n-2}{2} \rfloor$.*

PROOF. Let us consider the reachable states of the cross product automaton $A_n \times B_m$. (See Figure 3 for an example.) It is clear that $(1, 1)$ is mapped to $(2, 2)$ by a and to (n, m) by b . Also, (n, m) is a trap in $A_n \times B_m$. Since $1w \neq 1$ for any nonempty w neither in A_n nor in B_m , we get that $(1, 1)$ is the only reachable state of the form $(1, j)$ or $(i, 1)$ (i.e. the only reachable element of row 1 and column 1). Since $q \cdot_{B_m} b = q$ for $2 \leq q \leq m - 2$, and b induces a cycle on $\{2, \dots, n - 2\}$ in A_n , we get that any state of the form (p, q) with $2 \leq p \leq n - 2$ and $2 \leq q \leq m - 2$ is reachable if so is (p', q) for all $2 \leq p' \leq n - 2$.

Thus, reachability of $(2, 2)$ implies reachability of $(p, 2)$ for each $2 \leq p \leq n - 2$.

Also, by assumption $n \geq 5$, thus $3 \leq n - 2$ is an odd number, thus by definition of A_n , $3a = 3$. Moreover, $(3, 2)$ is reachable. Thus so is $(3, q)$ for each $2 \leq q \leq m - 2$ (by $q \cdot_{B_m} a = q + 1$ for $2 \leq q < m - 2$).

Hence, so is (p, q) for each $2 \leq p \leq n - 2$ and $2 \leq q \leq m - 2$ by the cycle $\{2, \dots, n - 2\}$ induced by b in A_n . Also, $(p, m - 2)a = (p, m - 1)$ for each odd $3 \leq p < n - 2$, thus such states $(p, m - 1)$ are also reachable. In particular, $(3, m - 1)$ is reachable, hence by $3 \cdot_{A_n} a = 3$ and $(m - 1)a = m$, so is $(3, m)$. Again by the cycle $\{2, \dots, n - 2\}$ induced by b in A_n we get that every state of the form (p, m) with $2 \leq p \leq n - 2$ is also reachable.

Now for the last two rows: by $(n - 2, p)a = (n - 1, p + 1)$ for $2 \leq p \leq m - 2$ we get that $(n - 1, p)$ is reachable for $3 \leq p \leq m - 1$. Also, $(n - 2, m)a = (n - 1, m)$ is reachable. By $(n - 1, p)b = (n, p)$ for $3 \leq p \leq m - 2$ we get each (n, p) with $3 \leq p \leq m - 2$ is reachable, and so are $(n, m - 2)a = (n, m - 1)$ and the trap state $(n, m - 1)a = (n, m)$ as well.

Overall, out of the nm states the following ones are not reachable:

- members of the first row or the first column, but $(1, 1)$ – that's $m + n - 2$ states;
- $(n - 1, 2)$ and $(n, 2)$ – that's two another states;
- $(p, m - 1)$ for even $2 \leq p \leq n - 2$ – that's $\lfloor \frac{n-2}{2} \rfloor$ states.

So far we have $nm - (n + m) - \lfloor \frac{n-2}{2} \rfloor$ reachable states, some of which might be indistinguishable. Note that for $K_n \cup L_m$, the accepting states are those of the form $(n - 1, q)$ with $3 \leq q \leq m$ and $(p, m - 1)$ for odd $3 \leq p \leq n - 2$ and $p = n - 1, n$.

Amongst the reachable states, $(n, m - 1)$, $(n - 1, m - 1)$ and $(n - 1, m)$ are equivalent (accepting only the empty word), with their merging making an additional reduction of 2 in the state complexity.

We claim that all the other states are pairwise distinguishable. It is clear that from any state different from (n, m) , either $(n - 1, q)$ or $(p, m - 1)$ is reachable for some $p \in [n]$ or $q \in [m]$, thus (n, m) is the only empty state.

First we show that final states are pairwise distinguishable (apart from the three one already marked for merging), by a case analysis. Let $(p, q) \neq (p', q')$ be final states, at least one of them being not a member of $\{(n - 1, m), (n - 1, m - 1), (n, m - 1)\}$.

- If $q = q' = m - 1$, then without loss of generality we can assume $p < p'$. Then $p < n - 1$ (otherwise $(p, q) = (n - 1, m - 1)$ and $(p', q') = (n, m - 1)$ which are already merged). Hence $ab^{n-2-p}a$ is accepted from (p, q) but not from (p', q') .

- If $p = p' = n - 1$, then we may assume $q < q'$. Then, a^{n-1-q} is accepted from (p, q) but not from (p', q') . (Note that $q'a^{n-1-q} \neq n - 1$ since $n - 1 - q > 1$ and K_n does not contain any word ending with at least two a 's.)
- If $p = n - 1$, $p' \neq n - 1$, $q' = m - 1$ **and** $q < m - 1$, then (p, q) accepts a^{m-1-q} . From state (p', q') , the word leads to a final state only if $p' = n - 2$ and $q = m - 2$. These states $(n - 1, m - 2)$ and $(n - 2, m - 1)$ are distinguishable by ba , mapping $(n - 1, m - 2)$ to the final state $(n, m - 1)$ and $(n - 2, m - 1)$ to the empty state.
- If $p = n - 1$, $p' \neq n - 1$, $q' = m - 1$ **and** $q = m$, then these states $(n - 1, m)$ and $(p', m - 1)$ with $p' \neq n$ are distinguishable since $(n - 1, m)$ accepts only ε , while $(p', m - 1)$ accepts $b^{n-2-p'}a$.

Next we show that non-final states are also pairwise distinguishable.

Observe that $(1, 1)$ is the only non-final nonempty state (p, q) for which $L_{(p,q)} \subseteq a\{a, b\}^*$ (i.e. all the other states accept some word beginning with b). Thus, $(1, 1)$ is distinguishable from any other state. We show that if $(p, q) \neq (p', q')$ are non-final states, then they are distinguishable, by another lengthy case analysis. Assume $p \leq p'$ and if $p = p'$, then $q < q'$ (that is, (p, q) lexicographically precedes (p', q')).

- If $p < p' < n - 1$ **and** $\{m - 2\} \neq \{q, q'\}$, then $b^{n-2-p}a$ is accepted only from p and $b^{n-2-p'}a$ is accepted only from p' in A_n , and by $\{m - 2\} \neq \{q, q'\}$, either q or q' does not accept any word of the form b^*a , thus (p, q) and (p', q') are distinguishable.
- If $p < p' < n - 1$ **and** $q = q' = m - 2$, **and** $p + n - p'$ is **even**, then $b^{n-2-p'}a$ takes (p', q') to $(n - 1, m - 1)$ and (p, q) to the state $(p + n - 2 - p', m - 1)$ which we already know to be distinguishable from $(n - 1, m - 1)$, hence so are (p, q) and (p', q') .
- If $p < p' < n - 1$ **and** $q = q' = m - 2$, **and** $p + n - p'$ is **odd**, then $b^{n-p}a$ takes (p, q) to $(3, m - 1)$ and (p', q') to $(n, m - 1)$ which are distinguishable, thus so are (p, q) and (p', q') .
- If $p < n - 1$ **and** $p' = n$, then (p', q') accepts all words of the form $b^*a^{m-1-q'}$, in $(n, m - 1)$. By $pb^{n-2-p}a = 3 < n - 1$ we get that (p, q) cannot accept $b^{n-2-p}a$ in a state merged with $(n, m - 1)$ and since we already know that final states are distinguishable, so are (p, q) and (p', q') .
- If $p = p'$ **and** $q < q'$, then a^{m-1-q} is accepted from (p, q) but not from (p', q') unless $p' = n - 2$ and $q = m - 2$. But also in this case, $(n - 2, m - 2)$ and $(n - 2, m)$ are distinguishable since so are $(n - 2, m - 2)b = (3, m - 2)$ and $(n - 2, m)b = (3, m)$.

Thus, after merging the states $(n - 1, m)$, $(n - 1, m - 1)$ and $(n, m - 1)$ we get a reduced automaton having $mn - (m + n) - 2 - \lfloor \frac{n-2}{2} \rfloor$ states, showing the claim (since $n \leq m$ can be assumed by symmetry).

We note that for symmetric difference the same construction works, with a minor change: in that case, $(n - 1, m - 1)$ is not a final state and is merged by the empty state.

References

References

- [1] Janusz A. Brzozowski, Galina Jirásková, Baiyu Li, and Joshua Smith. Quotient complexity of bifix-, factor-, and subword-free regular languages. In Pál Dömösi and Szabolcs Iván, editors, *AFL*, pages 123–137, 2011.
- [2] Yo-Sub Han and Kai Salomaa. State complexity of basic operations on suffix-free regular languages. *Theoretical Computer Science*, 410(2729):2537 – 2548, 2009.
- [3] Yo-Sub Han, Kai Salomaa, and Derick Wood. State complexity of prefix-free regular languages. In *Proceedings of DCFS06*, pages 165–176, 2006.
- [4] Yo-Sub Han, Kai Salomaa, and Sheng Yu. State complexity of combined operations for prefix-free regular languages. In Adrian Horia Dediu, Armand Mihai Ionescu, and Carlos Martí n Vide, editors, *Language and Automata Theory and Applications*, volume 5457 of *Lecture Notes in Computer Science*, pages 398–409. Springer Berlin Heidelberg, 2009.
- [5] H.J. Shyr. *Free Monoids and Languages*. & Tung Wu ta hsueh (Taipei, Taiwan). Department of Mathematics. Lecture notes. Institute of Applied Mathematics, National Chung-Hsing University, 1991.
- [6] Sheng Yu. State complexity of regular languages. *Journal of Automata, Languages and Combinatorics*, 6:221–234, 2000.

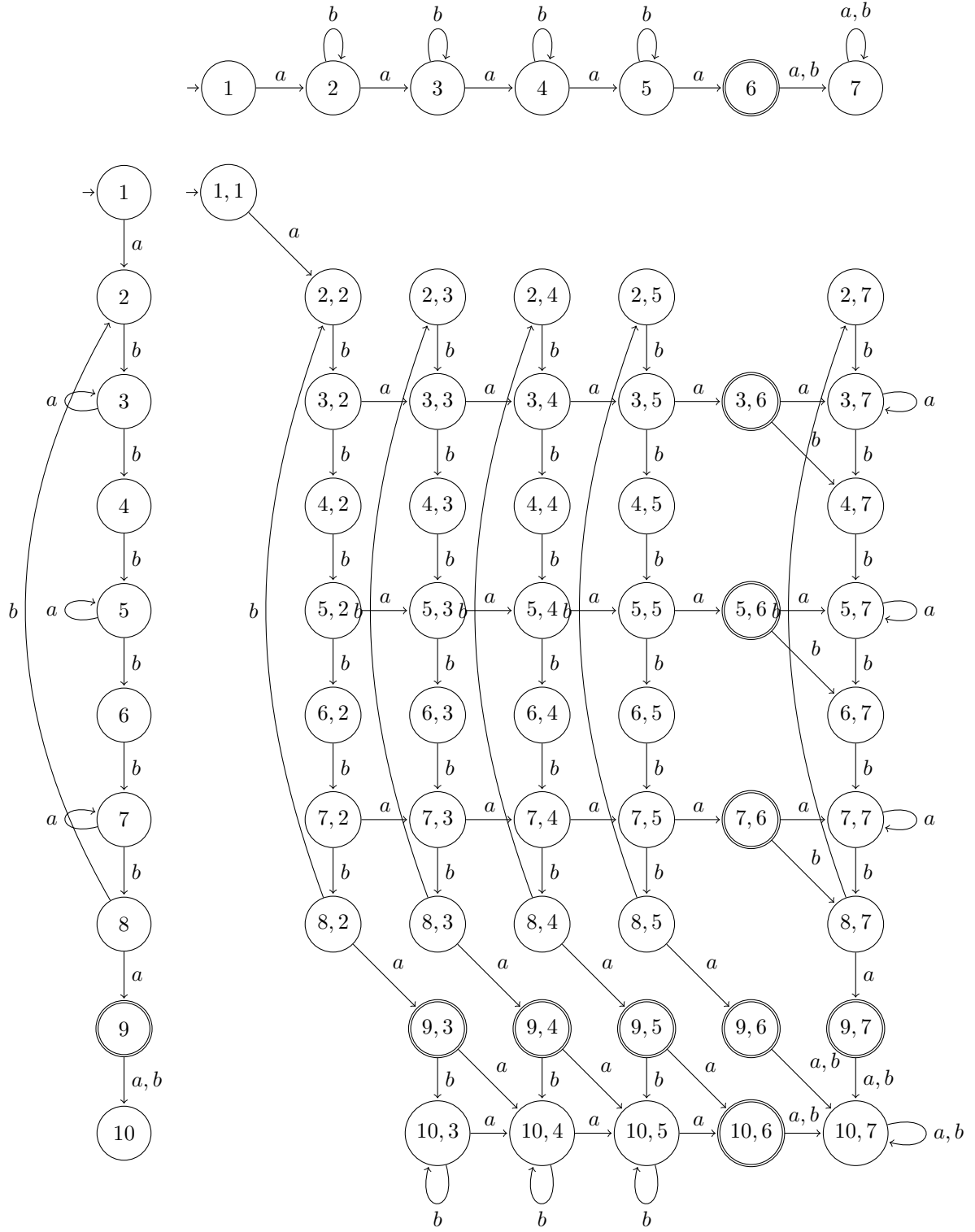


Figure 1: The automata A_{10} , B_7 and the reachable part of their cross product $A_{10} \times B_7$.